

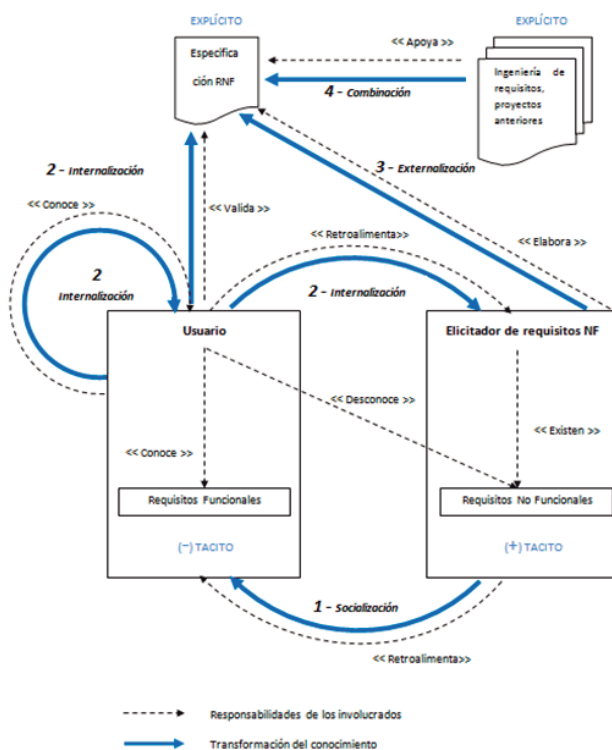
Transformación del conocimiento en el proceso de elicitación de requisitos no funcionales

SANDRA L. BUITRÓN¹
FRANCISCO J. PINO²

Resumen

La elicitación de requisitos (ER) implica un proceso de captura y entendimiento de las necesidades de los usuarios que serán usadas en las etapas siguientes del desarrollo del producto software. Dentro de las necesidades se deben incluir tanto las necesidades funcionales (RF) como las restricciones (conocidas como requisitos no funcionales, RNF), para así determinar la calidad del producto. En este sentido, su correcta consideración depende, en gran medida, de las prácticas para gestionar el conocimiento de los interesados (usuarios y elicitores). Usualmente los RNF no son tenidos en cuenta dentro de las organizaciones, generando impactos negativos en la calidad del producto. La intención del presente trabajo es determinar cómo se transforma el conocimiento durante el proceso de elicitación de requisitos no funcionales y establecer un conjunto de beneficios que las organizaciones pueden obtener al identificar adecuadamente este tipo de requisitos.

Palabras clave: ingeniería de software, elicitación de requisitos, gestión del conocimiento, requisitos no funcionales (RNF).



(1) Grupo IDIS, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca. Calle 2 No. 21 DN - 100. Popayán, Colombia. Correspondencia: Tel.: 57+8366330. Correo electrónico: sandrabr@unicauca.edu.co.

(2) Grupo IDIS, Facultad de Ingeniería Electrónica y Telecomunicaciones, Profesor Titular de la Universidad del Cauca, Calle 5 No. 4-70. Correspondencia: Tel: +5728209800 Ext. 2117. Correo electrónico: fjpino@unicauca.edu.co

Fecha de recepción: 16/04/2015 – Fecha de aceptación: 30/06/2015.

Abstract

The requirements elicitation (RE) is a process of capturing and understanding the needs of users that will be used in the following stages of software development product. Among the requirements should include both functional requirements (FR) and restrictions (known as non-functional requirements, NFR) to determine the quality of the product. Thus proper consideration depends largely on the knowledge management practices of stakeholders (users and eliciting). Usually RNF are not taken into account within organizations, creating negative impacts on the quality of the product. The intention of this paper is to determine how knowledge is transformed during elicitation of non-functional requirements, and establish a set of benefits that organizations can get to properly identify such requirements.

Keywords

Software Engineering, Requirements elicitation, Knowledge Management, Non-functional requirements (NFR).

1. Introducción

Para la construcción de productos de ingeniería del software se cuenta hoy en día con modelos que han sido definidos a lo largo de los años, conocidos como ciclos de desarrollo de software, con determinados enfoques, y que han sido aplicados desde diferentes perspectivas como la especificación, el diseño, la validación y la evolución [1]. Estos modelos definen las etapas para realizar el proceso de desarrollo de software; sin embargo, no indican de manera concreta cómo lograr la adhesión de cada propuesta al interior de la organización. Teniendo en cuenta las etapas que definen estas metodologías de desarrollo, se puede observar que la primera etapa siempre propone identificar los objetivos concretos, requisitos y características del entorno del negocio; en otras palabras, realizar lo que hoy conocemos como la elicitación de requisitos (ER). Este proceso de ER se considera como la base para que las etapas siguientes plasmen de manera adecuada y completa la(s) alternativa(s) de solución [2][3].

Revisando algunas definiciones sobre el proceso de elicitación de requisitos se tiene que este “es todo sobre aprender y entender las necesidades de los usuarios y de los interesados del proyecto con el objetivo principal de comunicar estas necesidades a los desarrolladores del sistema” [4]. Por otro lado, se define que la ER “es la fase principal enfocada en recopilar y analizar los requerimientos y objetivos deseados para el sistema desde diferentes puntos de vista (por ejemplo: clientes, usuarios, restricciones, entorno de operación del sistema, comercio, marketing y estándares etc. ...)”[2]. Además, bajo el estándar IEEE Std 830-1998 [5], el proceso de especificación de requisitos SRS (Software Requirements Specification) incluye, dentro de sus partes esenciales, información sobre interfaces externas, funciones, requisitos de desempeño, requisitos lógicos de base de datos, restricciones de diseño, atributos del software. Así mismo, para [6] la ER se ocupa de los orígenes de los requisitos del software y de cómo el ingeniero del mismo puede recolectarlos.

De acuerdo con estas definiciones, dentro de la elicitación de requisitos se tienen en cuenta dos elementos principales, los requisitos funcionales y los no funcionales [7]. Chung describe los requisitos no funcionales como características de usabilidad, flexibilidad, desempeño, operatividad y seguridad. Para [2] los requisitos funcionales son las acciones que debe realizar el software sin considerar las limitaciones físicas, mientras que los requisitos no funcionales definirán las propiedades ambientales y las restricciones de implementación relacionadas con el desempeño del mismo. Por otro lado, en [8] se indica que los requisitos no funcionales limitan el comportamiento y el desarrollo de un producto software, especificando los atributos que el sistema resultante debe tener. Además, en [9] se indica que la funcionalidad es lo que el sistema hace y su no funcionalidad o calidad se refiere a cómo se comporta el sistema frente a atributos observables como desempeño, reusabilidad, confiabilidad etc. El estudio de investigación realizado por [10] sobre el estado del proceso de elicitación de requisitos no funcionales permite comprobar que las técnicas utilizadas para realizar dicho proceso sobre RNF no están aún claras. De la misma manera, se afirma que “aunque existen técnicas bien desarrolladas para obtener requisitos funcionales, hay una falta de mecanismo de elicitación para NFR y no existe un consenso adecuado al respecto de los NFR” [11]. En este sentido,

es importante llevar a cabo una investigación sobre la identificación, análisis, documentación e integración de los requisitos no funcionales en el proceso de elicitación de requisitos, de tal manera que se pueda lograr un mayor cubrimiento y cohesión de dichos requisitos con el diseño de solución. Los requisitos no funcionales deben ser considerados como parte del proceso de desarrollo de software [7] y participan de manera determinante en la definición de las restricciones tanto para el desarrollo como para el proceso de implementación [11]; además, para lograr la calidad del producto de software se deben cumplir las características funcionales y las no funcionales, a fin de permitir cubrir de manera completa las expectativas de los interesados [12].

A partir de estos hallazgos, en este artículo se presenta: (i) una propuesta de cómo se transforma el conocimiento durante el proceso de elicitación de requisitos no funcionales, y (ii) un conjunto de beneficios que las organizaciones pueden identificar al realizar la elicitación de requisitos no funcionales. El objetivo de este trabajo es determinar cómo los requisitos no funcionales pueden ser incluidos de manera efectiva en el proceso de elicitación de requisitos bajo el enfoque de Gestión de Conocimiento. El artículo, además de esta introducción, se organiza de la siguiente manera: en la sección 2 se presentan los antecedentes de la temática, en la sección 3 se describe la propuesta de investigación y finalmente se exponen las conclusiones del trabajo realizado hasta el momento.

2. Antecedentes

En reportes como el que presenta Standish Groups's International se indican, entre otras, las tres razones principales que llevan al fracaso de los proyectos de desarrollo de software: en primer lugar, con un 15.9 %, la falta de participación de los usuarios; el segundo lugar lo ocupa la falta de soporte a la gestión ejecutiva del proyecto, con un 13.9 %; y en tercer lugar la definición clara de los requisitos, con un 13% [13]. Adicionalmente, [14] en su estudio confirman que frente al objetivo de lograr la calidad en las organizaciones, un 37.5 % de las no conformidades corresponde a requisitos no funcionales; entre otros resultados, se identifica que 5.88 % de las no conformidades hacen parte del proceso de definición de los requisitos de los stakeholders y un 12,94

% de las no conformidades se encuentran en el proceso de análisis de los requisitos del sistema. Por la relevancia descrita anteriormente, toma importancia la correcta y completa ejecución del proceso de elicitación de requisitos (ER) y, con mayor precisión, la ER enfocada en los RNF. En esta misma línea, para que el equipo de ER sea exitoso, se requiere, según [3], un profundo conocimiento del dominio de aplicación, IT y del proceso de elicitación; en otras palabras, una correcta combinación de conocimiento, recursos y procesos. La Gestión del Conocimiento es una disciplina importante que mediante un proceso sistemático permite crear y usar el conocimiento [15]; siendo el conocimiento una mezcla de experiencia, valores, información y "saber hacer", que sirve como marco para la incorporación de nuevas experiencias e información, y el cual resulta útil para la acción [16]. En esta línea, y debido a que la participación del usuario es un factor que contribuye en el éxito del proceso de elicitación de requisitos [17], el presente trabajo pretende abordar la administración del conocimiento de las necesidades del usuario y el conocimiento (técnico, del negocio, etc.) de los elicidores frente a la pregunta: ¿cómo apoyar la recolección de requisitos no funcionales en el proceso de elicitación de requisitos dentro de las organizaciones siguiendo un enfoque de Gestión del Conocimiento?

El aporte de este trabajo es la integración de procesos de creación y transformación del conocimiento con el proceso de elicitación de requisitos no funcionales, desde sus etapas más tempranas, de manera que se logra involucrar a los usuarios finales en actividades que permiten la obtención de los requisitos no funcionales y, de la misma manera, apoyar en dar mayor claridad y visibilidad a estas características de calidad del software dentro de los procesos de elicitación de requisitos que se lleven a cabo en las organizaciones.

3. Propuesta

Para abordar este problema y dar respuesta a la pregunta de investigación, se propone la creación de un marco de trabajo que permita incluir de manera efectiva los requisitos no funcionales en el proceso de elicitación de requisitos bajo el enfoque de Gestión de Conocimiento. En la creación de este marco es importante utilizar una

estrategia de investigación adecuada que permita abordar de una manera rigurosa y científica el problema, además de entender la transformación del conocimiento durante el proceso de ER no funcionales. En este sentido, a continuación de se presenta la estrategia de investigación que se utilizará para desarrollar la propuesta, luego el modelo SECI, posteriormente una descripción de cómo se transforma el conocimiento durante la ejecución del proceso de elicitación de requisitos no funcionales y, por último, algunos beneficios de incluir los requisitos no funcionales en el proceso de elicitación

3.1 Estrategia de investigación

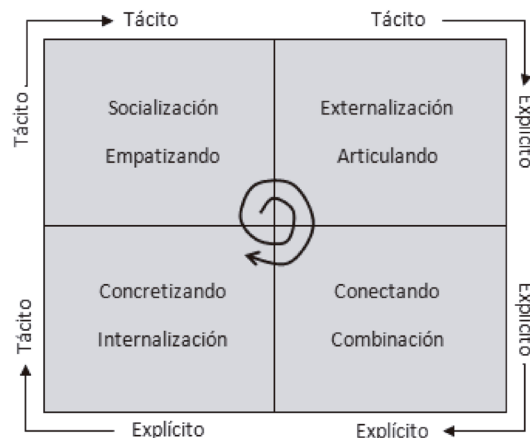
La estrategia de investigación utilizada en este trabajo es la investigación-acción multi-ciclo con bifurcación [18], a partir de la cual se desarrollarán los siguientes ciclos: conceptual, metodológico, de evaluación y de documentación. En el ciclo conceptual se identifica de manera concreta el problema dentro del contexto del proceso de desarrollo de software y se revisa la literatura buscando tanto características de calidad, mecanismos y técnicas para realizar el proceso de elicitación de RNF, así como conceptos relacionados con la Gestión de Conocimiento. En el ciclo metodológico se identifican y definen los componentes del marco de trabajo frente al proceso de elicitación de requisitos no funcionales y de la Gestión del Conocimiento, de manera que se logra una estructura específica de la alternativa de solución al problema de investigación. En los últimos dos ciclos de la estrategia, evaluación y documentación, se cubre el diseño del caso de estudio, su ejecución y análisis de resultados, así como la culminación de la documentación obtenida durante todo el proceso de investigación, finalizando con la divulgación de los resultados obtenidos.

3.2 Gestión del conocimiento y elicitación de requisitos no funcionales

La gestión del conocimiento es un proceso de responsabilidad de la alta gerencia, la cual debe propiciar ambientes que atraigan y sostengan la creación, uso y accesibilidad del nuevo conocimiento, con el fin de ahorrar tiempo, mejorar la productividad, propiciar la innovación y apoyar a la gestión de los procesos [19]. El proceso de

creación de conocimiento en las organizaciones se da a través de la interacción entre conocimiento explícito y tácito [20]. El modelo SECI es propuesto por [20] para la creación de conocimiento, el cual define cuatro formas para realizar “conversión de conocimiento”: socialización, externalización, combinación e internalización, como lo muestra la Figura 1.

Figura 1. El proceso SECI (tomada de [20])



Fuente: elaboración propia.

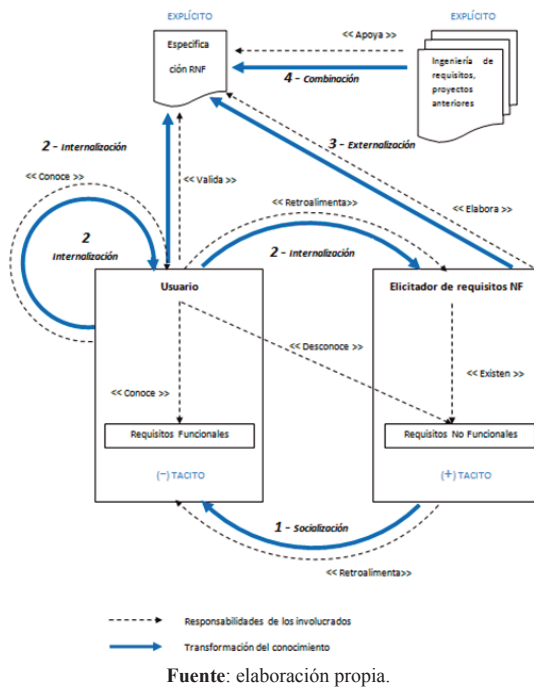
La Gestión de Conocimiento permitirá aportar al proceso de consecución, análisis y definición de los requisitos no funcionales, de manera que considere el conocimiento que está en los recursos (cliente, elicitor, activos de conocimiento) durante la ejecución del proceso de elicitación. De esta manera estaremos acercándonos, con mayor precisión, a lograr la calidad del producto software. Según el Software Engineering Institute (SEI), para lograr la calidad en los productos de software se deben combinar las siguientes tres dimensiones organizacionales: las personas, los métodos y procedimientos, y las herramientas, siendo los procesos utilizados por la organización los que permiten evolucionar e incorporar los conocimientos de cómo hacer mejor las cosas [21].

3.3 Transformación del conocimiento en el proceso de elicitación de requisitos no funcionales

La Figura 2 nos muestra una descripción sobre cómo se da la transformación del conocimiento durante el proceso de elicitación de RNF. Al iniciar el proceso es el elicitor quien debe contextualizar al usuario sobre lo que son, la

importancia y las posibles maneras de identificar los RNF para el producto software a desarrollar (socialización). En este momento, el conocimiento tácito del usuario es menor frente al grado de conocimiento tácito del elicitor en relación con los RNF. El usuario omite estos aspectos durante el proceso de elicitación [3] [12]; por lo tanto, desconoce los RNF, mientras que el elicitor ya cuenta con determinada formación y experiencia sobre la existencia de los RNF y la importancia de determinarlos.

Figura 2. Transformación del conocimiento en el proceso de elicitación de requisitos no funcionales



Fuente: elaboración propia.

A medida que se realiza esta transferencia de conocimiento, el usuario adquiere nueva información, de modo que empieza a entender e interiorizar los conceptos, la importancia, los propósitos y las formas de identificar los RNF (internalización en el usuario). En esta dinámica de retroalimentación, el elicitor podrá utilizar la información obtenida para ir concretando los RNF determinados por el usuario (internalización en el elicitor). Al terminar estas actividades iniciales de elicitación, el elicitor podrá elaborar un primer documento de especificación de los RNF del producto (externalización), el cual podrá ser complementado con conocimientos de apoyo dispuestos en otros documentos de ingeniería de requisitos o documentos de RNF de

otros proyectos (combinación). Cuando se tiene este documento, se comparte con el usuario para buscar su validación y aprobación, a fin de que pueda ser considerado como documento final del proceso de elicitación de requisitos NF.

3.4 Beneficios de la propuesta para las organizaciones

Esta propuesta podría apoyar a las organizaciones en la consecución de beneficios que, desde la propia experiencia, se han identificado a través del trabajo llevado a cabo durante 15 años en el campo de administración de procesos en diferentes empresas de la industria de software nacional; entre los cuales están:

- *Mayor concientización* de los usuarios frente a la importancia y relevancia de estos aspectos no funcionales en el momento de la recolección y posterior aprobación de la especificación del sistema.
- Al lograr una mayor conciencia de estos aspectos se irá fortaleciendo el proceso de elicitación en las organizaciones, de manera que se empiece a incluir lo no funcional como factor necesario para lograr la calidad.
- Los usuarios y demás interesados alcanzarán un *mayor grado de confianza* con respecto a la información crítica para el negocio que genere el sistema, debido a que podrán verificar estos resultados a través de los elementos no funcionales que sean identificados e incluidos en el momento de la elicitación.

- Disminución de los eventos reportados al grupo de soporte de las organizaciones, como consecuencia de la capacidad del producto para indicar al usuario final sobre el motivo de los errores y la ruta para solucionarlos.

- Se podrá generar un plan de pruebas mucho más robusto, que permita a las organizaciones evidenciar el comportamiento del sistema en un escenario de pruebas, generando mayor tranquilidad en los usuarios con respecto a su comportamiento en términos de tiempos de respuesta y de los componentes de arquitectura (base de datos, canales de comunicación, servidores, equipos cliente etc.). Esto es, *obtener información cuantitativa y cualitativa oportunamente* sobre el desempeño del producto, permitiendo la toma de decisiones con

respecto a la futura arquitectura que se utilizará en el entorno productivo.

- Todos los aspectos no funcionales que se logren recolectar durante la elicitación permitirán un *mayor volumen de información relevante para el posterior diseño de la arquitectura del producto*. Estos aspectos podrán ser tenidos en cuenta al momento de tomar decisiones referentes al dimensionamiento de las necesidades de hardware (conocido como *sizing*) por las organizaciones y su departamento de tecnología.

- Esta información no funcional podría llevar a los administradores del sistema a definir un plan de monitoreo del sistema bajo el entorno productivo, de manera que se logre disminuir la atención de emergencias y correctivos frente a la disponibilidad del servicio que presta el producto software; esto es, se podrá brindar un servicio de soporte tecnológico de tipo preventivo a la organización.

- Cada uno de estos aspectos no funcionales que se incluyan en el plan de monitoreo, y que, por lo tanto, ameriten ser medidos y controlados, podrían apoyar en la definición de los indicadores del servicio de soporte del departamento de tecnología, toda vez que estas mediciones sobre el producto software se convierten en bases estadísticas para cumplir con la responsabilidad de seguimiento y mantenimiento a los sistemas de información como activos tecnológicos de las organizaciones.

- Esta inevitable relación entre el proceso de elicitación de requisitos no funcionales y el proceso de diseño de la arquitectura del producto, podrían beneficiar en la convergencia y cohesión de esfuerzos de los integrantes del departamento de tecnología de las organizaciones, de modo que se mejore la calidad del servicio como área de apoyo dentro de la organización.

- Las interacciones y participación de diferentes niveles y tipos de conocimiento (funcional, técnico y de negocio) durante el proceso de elicitación de requisitos no funcionales, permite a los ingenieros involucrados en este proceso fortalecer continuamente su conocimiento integral del negocio, así como afianzar su conocimiento técnico, ejecutando procesos de ingeniería mucho más maduros en pro de los objetivos y metas organizacionales.

- La definición clara de los aspectos no funcionales

referentes al producto de software, permitirá un análisis, diagnóstico y atención más rápidos y precisos ante los eventos de emergencia reportados por los usuarios, logrando un mejor tiempo de respuesta del servicio del departamento de tecnología; esto, debido a que la elicitación de requisitos no funcionales permite al final obtener la correlación de estos aspectos no funcionales con la funcionalidad del producto, haciendo las veces de mapa de ruta para la atención de eventos.

- El conocimiento de quienes participan de manera activa en procesos de elicitación de requisitos no funcionales, tanto usuarios como ingenieros de software, podrán obtener mayor habilidad frente al proceso, lo que creará un hábito organizacional de inclusión de los requisitos no funcionales en sus proyectos de desarrollo de software.

4. Conclusiones

La literatura encontrada hasta el momento permite confirmar la importancia de realizar mayores esfuerzos en el tema de elicitación de requisitos no funcionales. En esta literatura se evidencia que las características de calidad del software requieren ser visualizadas de forma más clara ante las organizaciones que realizan proyectos de desarrollo de software; por otro lado, la Gestión del Conocimiento ofrece aspectos que pueden ser usados para incluir los requisitos no funcionales en el proceso de elicitación de requisitos, ya que abarca todas las fuentes posibles de conocimiento dentro de las organizaciones, tales como los procesos, los interesados y la tecnología existente. Es así como este escenario de investigación posibilita la generación de una propuesta sobre el cómo realizar la elicitación de requisitos no funcionales desde la perspectiva de Gestión de Conocimiento.

Estos esfuerzos podrán ser realizados por las organizaciones a medida que se brinden mecanismos prácticos que faciliten su apropiación dentro de sus procesos de desarrollo de software. La Gestión de Conocimiento es un enfoque que surge como alternativa aplicable para la mejora de procesos relacionados con el conocimiento de los usuarios finales (stakeholders), dentro del contexto de la elicitación de requisitos. En este orden de ideas, en este trabajo se ha presentado un modelo de cómo se transforma el conocimiento durante la elicitación de requisitos no funcionales (y algunos beneficios de considerar este tipo de

requisitos). Como trabajo futuro se abordará la definición de un marco de trabajo que permita incluir de manera efectiva los requisitos no funcionales en el proceso de elicitación y esté basado en el modelo definido.

Agradecimientos

Este trabajo ha sido financiado por el proyecto "Marco de trabajo para la elicitación de requisitos no funcionales basado en la gestión del conocimiento" (Vicerrectoría de Investigaciones de Unicauca - VRI ID 4354). Los autores agradecen a: (i) Brenda L. Flores Ríos por su valioso aporte y sugerencias; y (ii) la Universidad del Cauca, donde Sandra Buitrón cursa la Maestría en Computación y Francisco J. Pino trabaja como profesor titular ●

Referencias

- [1] Mohammed, N., Munassar, A. & Govardhan, A. (2010). A Comparison Between Five Models Of Software Engineering. *IJCSI International Journal of Computer Science Issues* 7, 5, 94 - 101.
- [2] Pandey, D., Suman, U. & Ramani, K. (Oct., 2010). *An Effective Requirement Engineering Process Model for Software Development and Requirements Management*. 2010 Int. Conf. Adv. Recent Technol. Commun. Comput. 287 - 291.
- [3] Hofmann, H. F. & Motors, G. (Aug., 2001). *Requirements Engineering as a Success Factor in Software Projects*.
- [4] Zowghi, D. & Coulin, C. (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools. *Eng. Manag. Softw. Requir.* 19 - 46.
- [5] Engineering, S. & Committee, S. (1998). *IEEE Recommended Practice for Software Requirements Specifications*.
- [6] IEEE Computer. (2014). *Guide to the Software Engineering Body of Knowledge Version 3.0 (SWEBOK Guide V3.0)*.
- [7] Chung, L., Cesar, J. & Leite, P. (2009). *On Non-Functional Requirements in Software*. pp. 363 - 379.
- [8] Casamayor, A., Godoy, D. & Campo, M. (Apr., 2010). Identification of non-functional requirements in textual specifications: A semi-supervised learning approach. *Inf. Softw. Technol.*, 52, 4, 436 - 445.
- [9] Franch, X. & Botella, P. (1998). *Putting Non-Functional Requirements into*, 1158.
- [10] Serna-Montoya, E. (2011). *Estado actual de la investigación en requisitos no funcionales* 1, 2126, 225 - 246.
- [11] Mijanur Rahman, M. & Ripon, S. (2013). Elicitation and Modeling Non-Functional Requirements – A POS Case Study. *Int. J. Futur. Comput. Commun.* 2, 5, 485 - 489.
- [12] Cysneiros, L. M. & Yu, E. (2004). *Chapter # Non-functional requirements*.
- [13] Standish Group's International. (2002). *The standish group report* ©. 1 - 8.
- [14] Garzás, J., Pino, F. J., Piattini, M. & Manuel, C. (2013). *A maturity model for the Spanish software industry based on ISO standards*.
- [15] Nonaka, I. & Teece, D. J. (1998). *Managing Industrial Knowledge*.
- [16] Davenport, B. T. H., Prusak, L. & Webber, A. (2011). *Working Knowledge: How Organizations Manage What They Know*. 1 - 15.
- [17] Adhavji, N. A. H. M., El Emam, K. & Madhavji, N. H. (March, 1995). *A Field Study of Requirements Engineering Practices in Information Systems Development A Field Study of Requirements Engineering Practices in Information Systems Development*.
- [18] Pino, F. J., Piattini, M. & Travassos, G. H. (2013). *Managing and developing distributed research projects in software engineering by means of Gestión y desarrollo de proyectos de investigación distribuidos en ingeniería del software por medio de investigación-acción*. pp. 61 - 74.
- [19] Milena, C. & Raza, H. (2014). *Técnica para el análisis de requisitos de software basada en la gestión del conocimiento, para la empresa de desarrollo de software Kruger*.
- [20] Nonaka, I., Toyama, R. & Konno, N. (2000). *SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation*, vol. 33, 5 - 34.
- [21] Chrissis, M. B. & Torralba, J. (2009). *CMMI® Guía para la integración de procesos y la mejora de productos*.